

Windows Programming

Visual C++ MFC Programming

Lecture 12

김예진

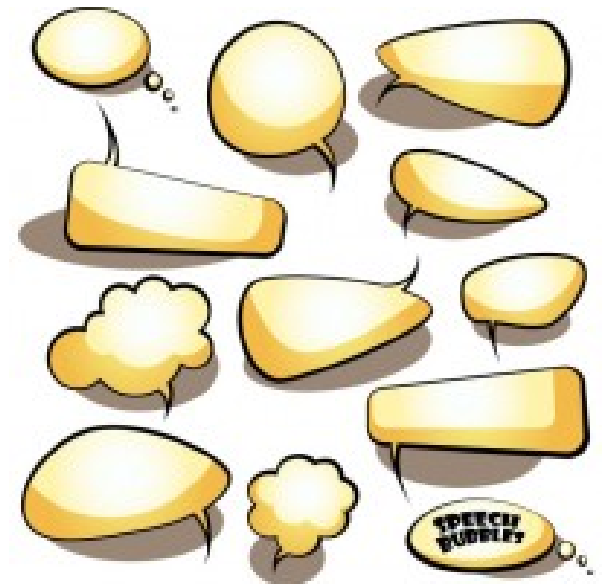
Dept. of Game Software

Notices

- 03/07: 502 → 501 등록 이동
- 03/21: HW 1 (Due: 03/28)
- 04/09: HW 2 (Due: 04/16)
- 04/25: Midterm (실습, ~75 min, 강의록 1~7)
- 05/14: HW 3 (Due: 05/21)
- 06/04: HW 4 (Due: 06/11)
- 06/13: Final (실습, ~75 min., 강의록 전체)
- 06/18: 보강
 - 501: 13:00~15:00
 - 502: 15:00~17:00

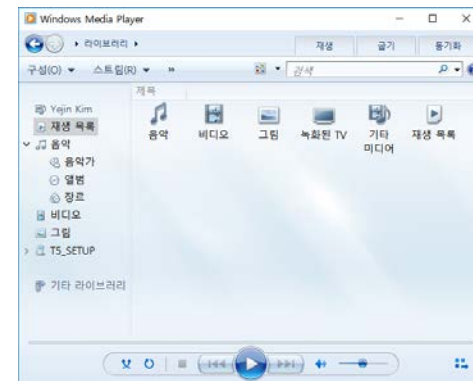
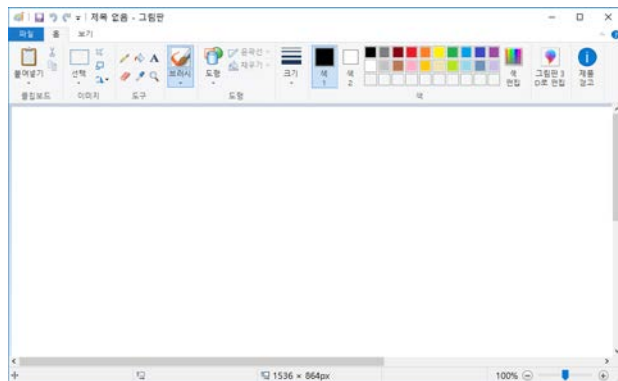
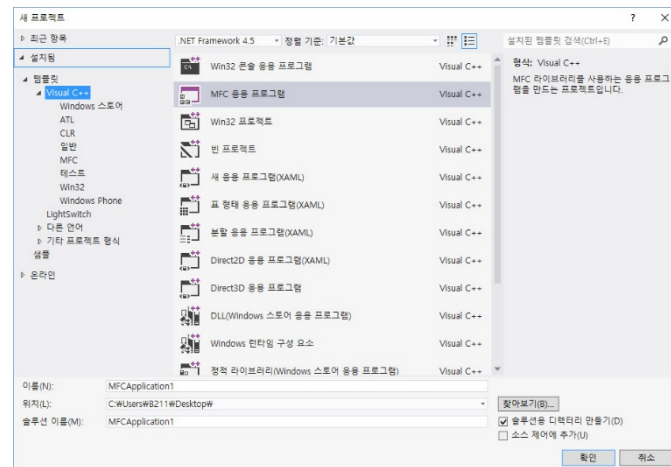
Plan

- 모드형 대화상자(modal dialog box)
- 비모드형 대화상자(modeless dialog box)
- 공통 대화 상자(common dialog box)



Dialog Box

- 대화 상자(dialog box)
 - 다양한 control을 포함하고 있는 일종의 윈도우
 - 사용자로부터 입력을 받거나 정보를 출력
- 대화 상자 기반 응용 프로그램
 - 대화 상자가 main 윈도우 역할을 하는 응용 프로그램



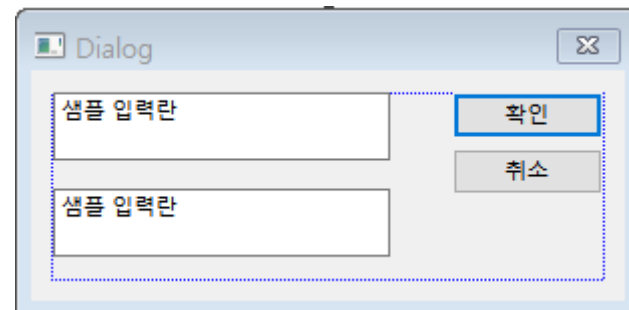
Dialog Box

- 대화 상자 종류
 - 모드형 대화 상자(modal dialog box)
 - 대화 상자를 닫지 않으면 응용 프로그램이 더 이상 진행할 수 없음
 - Stack memory에 생성: 대화 상자를 종료 전 data를 관리해야함
 - 비모드형 대화 상자(modeless dialog box)
 - 대화 상자를 닫지 않더라도 응용 프로그램이 계속 진행할 수 있음
 - Heap memory에 생성: 대화 상자를 종료하지 않고 data 관리 가능
- 대화 상자 생성
 - 대화 상자는 운영체제 내부적으로 template 형태로 존재
 - 대화 상자 자체와 포함된 control에 대한 모든 정보를 가지고 있음
 - 대화 상자 template 생성 과정
 1. Resource view나 text 편집기를 이용하여 resource script(*.RC) 작성
 2. Build 과정에서 resource compiler가 binary file(*.RES)로 생성
 3. Link 과정에서 실행 파일에 포함

Dialog Box

- 대화 상자 template 생성
 - Resource script(*.RC)를 text 편집기로 직접 작성

```
IDD_DIALOG1 DIALOGEX 0, 0, 197, 71
STYLE DS_SETFONT | DS_MODALFRAME | DS_FIXEDSYS |
WS_POPUP | WS_CAPTION | WS_SYSMENU
CAPTION "Dialog"
FONT 8, "MS Shell Dlg", 400, 0, 0x1
BEGIN
    DEFPUSHBUTTON    "확인", IDOK, 140, 7, 50, 14
    PUSHBUTTON       "취소", IDCANCEL, 140, 24, 50, 14
    EDITTEXT         IDC_EDIT1, 7, 7, 112, 21, ES_AUTOHSCROLL
    EDITTEXT         IDC_EDIT2, 7, 36, 112, 21, ES_AUTOHSCROLL
END
```

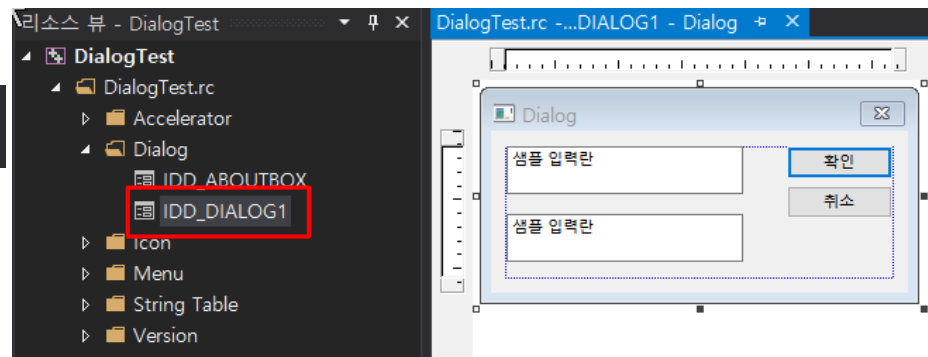


- Resource view에서 작성
 - Dialog → [RMB] Dialog 삽입 → [도구 상자] Control 추가
 - 대화 상자 편집기 사용



- 서식(Format) 메뉴 사용

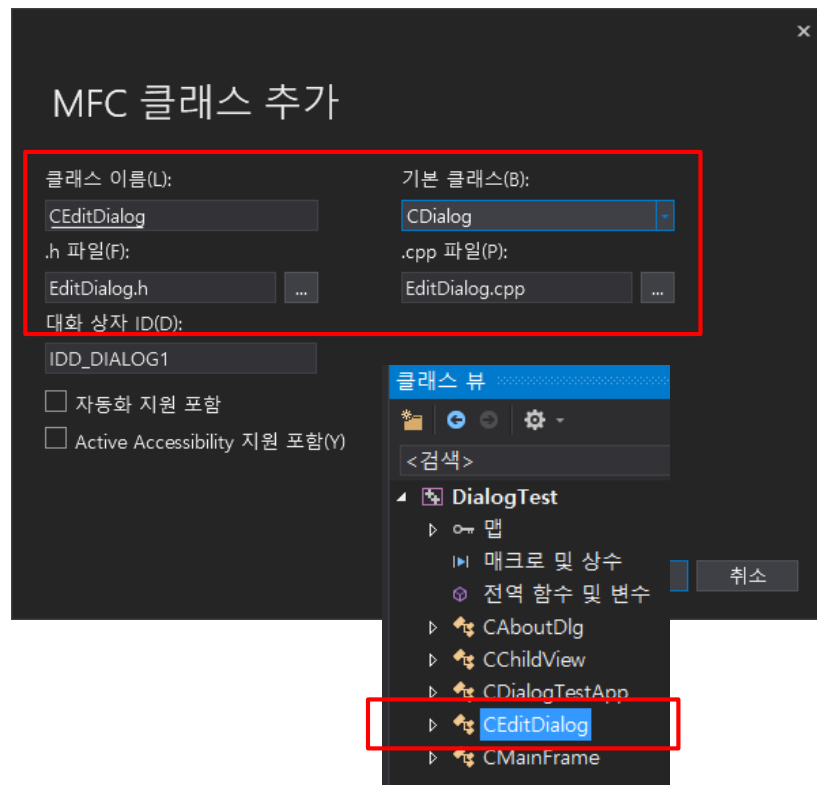
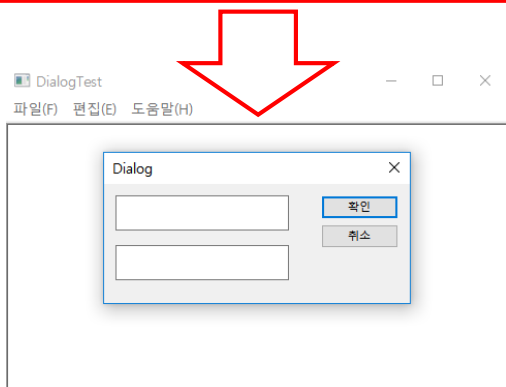
디버그(D) 팀(M) 서식(O) 도구(T) 테스트(S)



Modal Dialog Box

- 모드형 대화 상자(modal dialog box) 생성
 1. 대화 상자 resource 생성: 직접 코딩 or resource view 사용
 2. CDialog (or CDialogEX) object class 생성
 - 리소스 뷰 → 대화 상자 ID → [RMB] Dialog → 클래스 추가
 3. CDialog::DoModal() 호출

```
void CChildView::OnLButtonDown(UINT nFlags, CPoint point)
{
    CEditDialog Dlg;
    Dlg.DoModal();
}
```



Modal Dialog Box

- CDialog class 가상 함수 사용
 - 가상 함수 추가: 클래스 마법사 → 가상 함수

```
virtual BOOL CDialog::OnInitDialog();
```

- 호출 시점: WM_INITDIALOG 발생시
- 용도: control을 초기화하거나 keyboard focus를 변경

```
virtual void CDialog::OnOK();
```

- 호출 시점: IDOK button을 누를 때
- 용도: control의 값을 읽거나 값의 타당성 여부를 검사한 후 대화 상자 종료

```
virtual void CDialog::OnCancel();
```

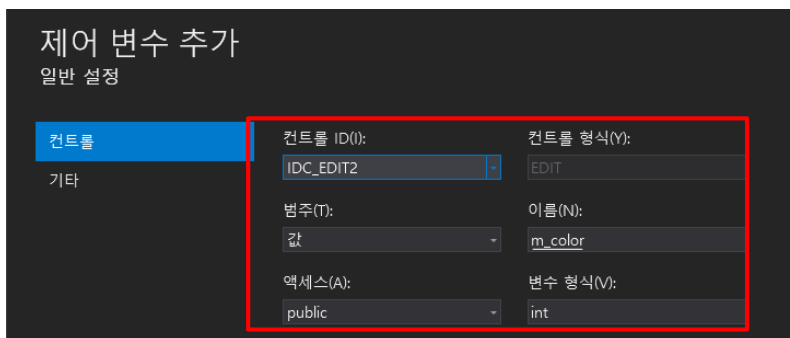
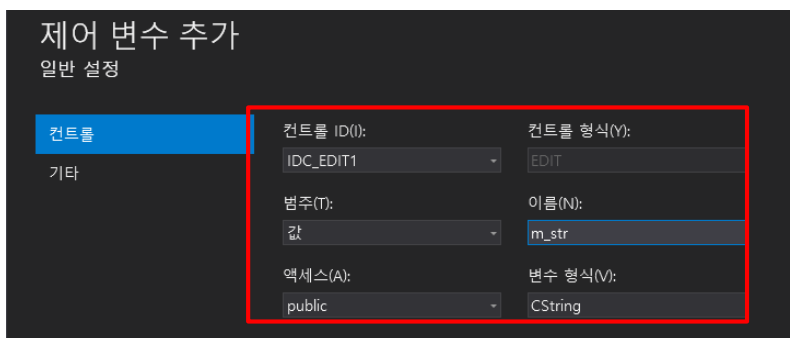
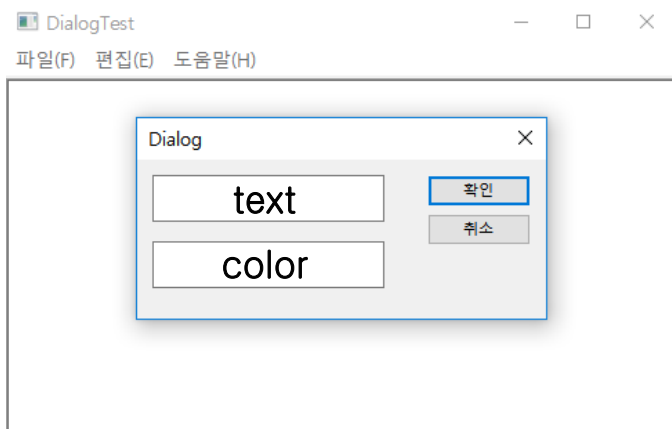
- 호출 시점: IDCANCEL 버튼을 누를 때
- 용도: 대화 상자 종료

```
void CEditDialog::OnOK()  
{  
    EndDialog(IDOK);           // 확인(OK)로 대화 상자 종료  
}
```

```
void CEditDialog::OnCancel()  
{  
    EndDialog(IDCANCEL);       // 취소(cancel)로 대화 상자 종료  
}
```


Modal Dialog Box

- 대화 상자의 control 사용
 - 각 control마다 알맞은 data type 변수 추가
 - 사용 예
 - IDC_EDIT1 → text 입력(CString)
 - IDC_EDIT2 → color 입력(int)



Modal Dialog Box

- 대화 상자의 control 사용
 - 변수 update

```
class CEditDialog : public CDialog
{
public:
    CString m_str;
    int m_color;
}

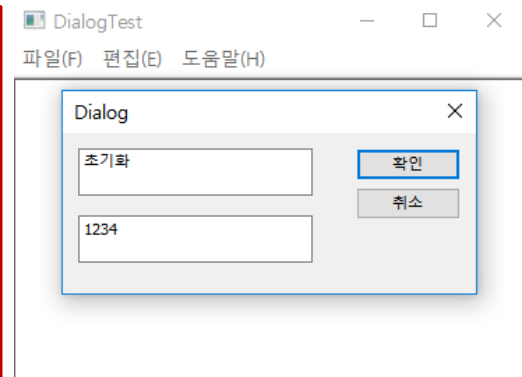
BOOL CEditDialog::OnInitDialog()
{
    CDialog::OnInitDialog();

    SetDlgItemText(IDC_EDIT1, _T("초기화")); // IDC_EDIT1에 text를 전달
    SetDlgItemInt(IDC_EDIT2, 1234); // IDC_EDIT2에 color 값을 전달

    return TRUE;
}

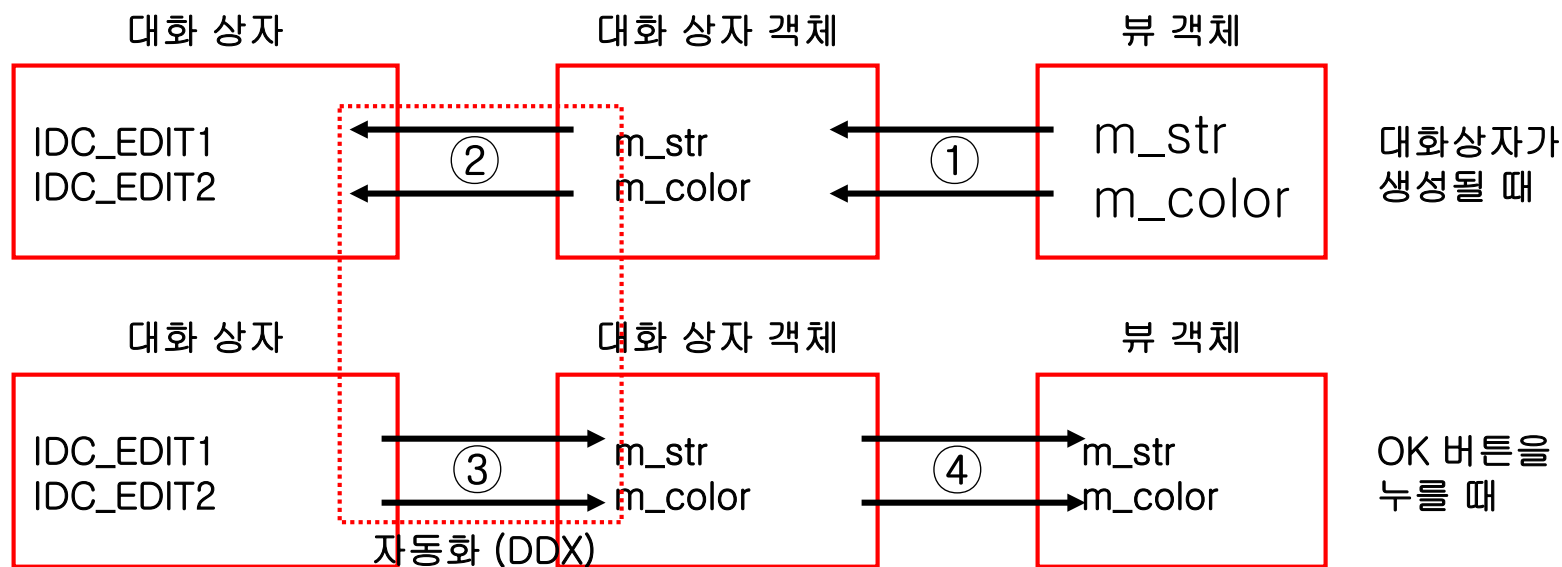
void CEditDialog::OnOK()
{
    GetDlgItemText(IDC_EDIT1, m_str); // IDC_EDIT1에서 text를 얻음
    m_color = GetDlgItemInt(IDC_EDIT2); // IDC_EDIT2에서 color 값을 얻음

    CDialog::OnOK();
}
```



Modal Dialog Box

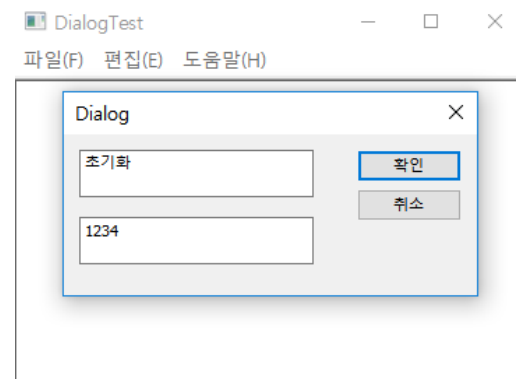
- 대화 상자의 control data 이동
 - Dialog Data eXchange(DDX) 사용
 - 대화 상자의 data \leftrightarrow 대화 상자의 변수로 자동 이동
 - DDX를 사용한 다른 객체(예: CChildView)로의 data 이동 원리



Modal Dialog Box

- 대화 상자의 control data 이동: DDX 사용
 1. Control의 값을 저장할 변수 선언
 2. DoDataExchange() 함수를 자신의 프로그램에 맞게 재정의
 - DDX_* 매크로 사용

```
CEditDialog::CEditDialog() {  
    m_str = "초기화";  
    m_color = 1234;  
}  
  
void CEditDialog::DoDataExchange(CDataExchange* pDX) {  
    CDialog::DoDataExchange(pDX);  
    DDX_Text(pDX, IDC_EDIT1, m_str);  
    DDX_Text(pDX, IDC_EDIT2, m_color);  
}  
  
void CEditDialog::OnInitDialog() {  
    CDialog::OnInitDialog();  
    UpdateData(FALSE);           // 변수로 부터 control의 값을 초기화  
    return TRUE;  
}  
  
void CEditDialog::OnOK() {  
    UpdateData(TRUE);           // Control에 입력된 값을 변수 값에 적용  
    CDialog::OnOK();  
}
```



Modal Dialog Box

- 대화 상자의 control data 타당성 검사
 - Dialog Data Validation(DDV) 사용
 - 대화 상자의 control에 입력한 data의 타당성 여부를 자동으로 검사
 - DDV_* 매크로 사용

```
void CEditDialog::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
    DDX_Text(pDX, IDC_EDIT1, m_str);
    DDV_MaxChars(pDX, m_str, 10);
    DDX_Text(pDX, IDC_EDIT2, m_color);
    DDV_MinMaxInt(pDX, m_color, 0, 255);
}
```

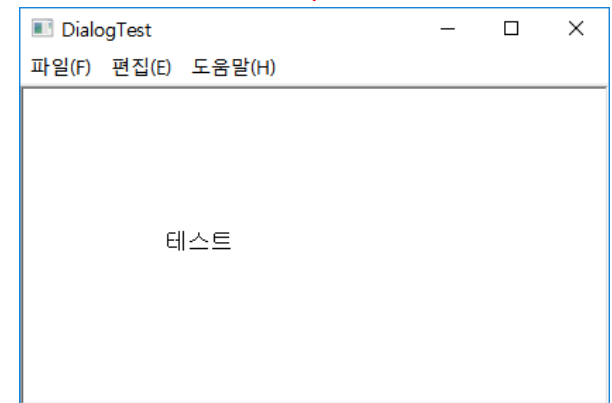
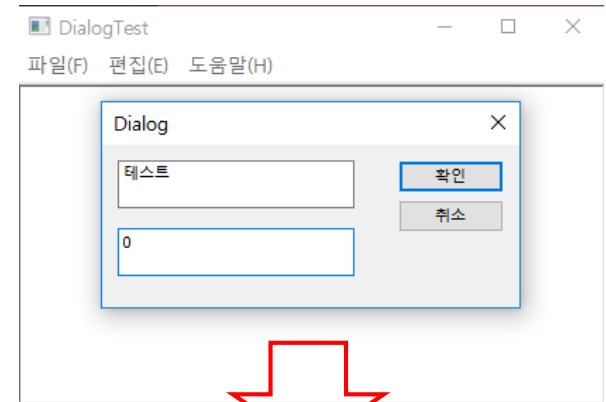
Modal Dialog Box

- 대화 상자의 control data를 사용자 영역에 출력
 - CChildView에서 control의 값을 저장할 변수 선언
 - CEditDialog 변수 값 → CChildView 변수 값 이동

```
class CChildView : public CWnd {
public:
    CString m_str;
    int m_color;
}

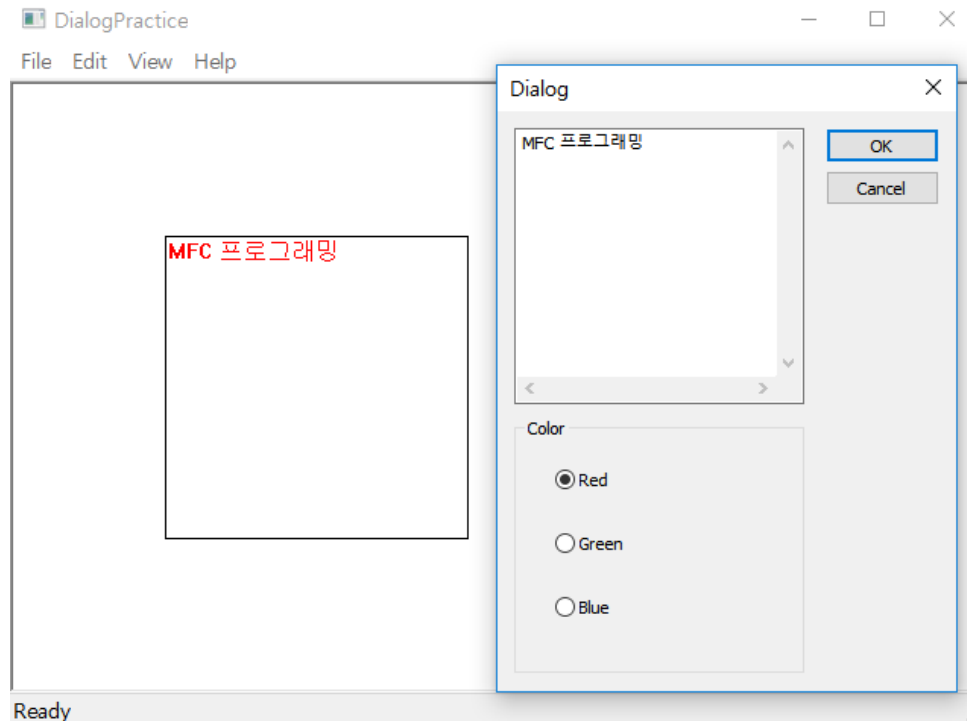
void CChildView::OnPaint()
{
    CPaintDC dc(this);
    COLORREF color = RGB(m_color, m_color, m_color);
    dc.SetTextColor(color);
    dc.DrawText(m_str, CRect(100, 100, 200, 130), DT_LEFT);
}

void CChildView::OnLButtonDown(UINT nFlags, CPoint point)
{
    CEditDialog Dlg;
    if (Dlg.DoModal() == IDOK) { // 확인 버튼을 누르면
        m_str = Dlg.m_str;
        m_color = Dlg.m_color;
    }
    Invalidate();
}
```



Modal Dialog Box

- 연습: Text 입력과 색상 지정이 가능한 모드형 대화 상자
 - Text 입력: edit box
 - 색상(R, G, B): radio button 3개
 - 대화 상자 호출: LMB click
 - OK 버튼: 대화 상자 종료 후 사용자 영역에 입력한 text 표시



Modeless Dialog Box

- 비모드형 대화 상자(modeless dialog box) 생성
 - Resource를 만들고, class를 만드는 방법은 모드형과 동일
 - 생성시 CDialog::DoModal() 대신 CDialog::Create()를 사용
 - 종료시 CDialog::EndDialog() 대신 CWnd::DestroyWindow()를 호출

```
class CChildView : public CWnd {
public:
    CEditDialog *m_pDlg;
}

CChildView::CChildView() {
    m_pDlg = NULL;
}

void CChildView::OnRButtonDown(UINT nFlags, CPoint point) {
    if (m_pDlg != NULL)
        m_pDlg->SetFocus();
    else {
        m_pDlg = new CEditDialog();
        m_pDlg->m_str = m_str;
        m_pDlg->m_color = m_color;
        m_pDlg->m_pView = this;
        m_pDlg->Create(IDD_DIALOG1);
        m_pDlg->ShowWindow(SW_SHOW);
    }
}
```

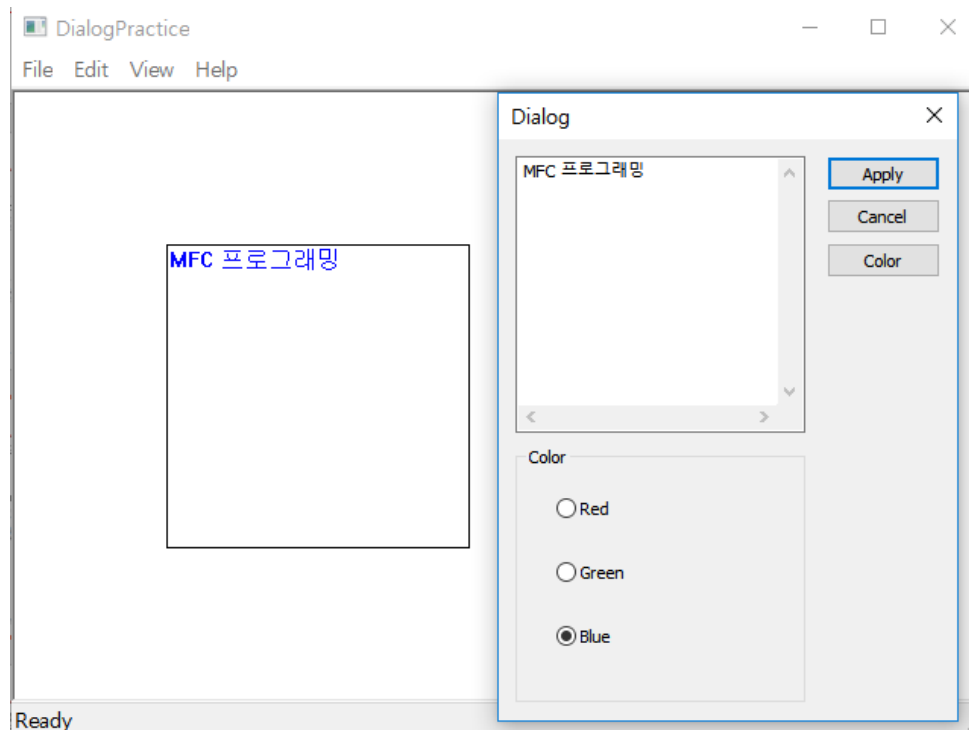
```
class CEditDialog : public CDialog {
public:
    CChildView m_pView;
}

void CEditDialog::OnOK() {
    UpdateData(TRUE);
    m_pView->m_str = m_str;
    m_pView->m_color = m_color;
    m_pView->Invalidate();
}

void CEditDialog::OnCancel()
{
    DestroyWindow();
    m_pView->m_pDlg = NULL;
}
```


Modeless Dialog Box

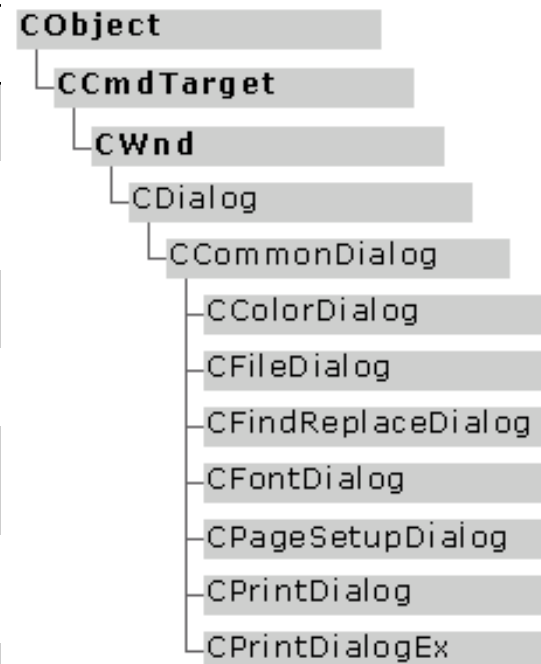
- 연습: Text 입력과 색상 지정이 가능한 비모드형 대화 상자
 - Text 입력: edit box
 - 색상(R, G, B): radio button 3개
 - 대화 상자 호출: RMB click
 - Apply 버튼: 사용자 영역에 입력한 text 표시 (*대화 상자 종료 없이)



Common Dialog Box

- 공통 대화 상자(common dialog box)
 - 응용 프로그램에서 흔히 필요한 대화상자
 - MFC 공통 대화 상자 class

Class	용도	API Function
CColorDialog	색상 선택	ChooseColor
CFileDialog	파일 열기 또는 저장	GetOpenFileName GetSaveFileName
CFindReplaceDialog	찾기 또는 바꾸기	FindText ReplaceText
CFontDialog	폰트 선택	ChooseFont
CPageSetupDialog	페이지 설정(페이지 크기, 방향, 페이지 여백 등)	PageSetupDlg
CPrintDialog	인쇄 설정(프린터, 인쇄 범위 등)	PrintDlg
CPrintDialogEx	인쇄 설정(프린터, 인쇄 범위 등)	PrintDlgEx

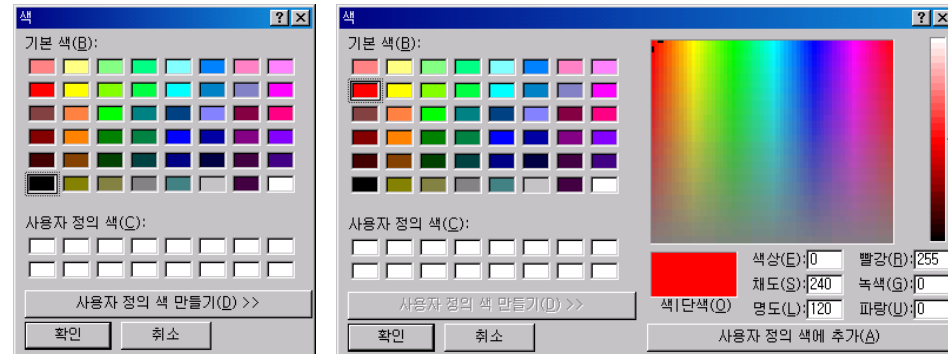


Common Dialog Box

- CColorDialog

```
CColorDialog dlg;  
dlg.DoModal();  
COLORREF color = dlg.GetColor();
```

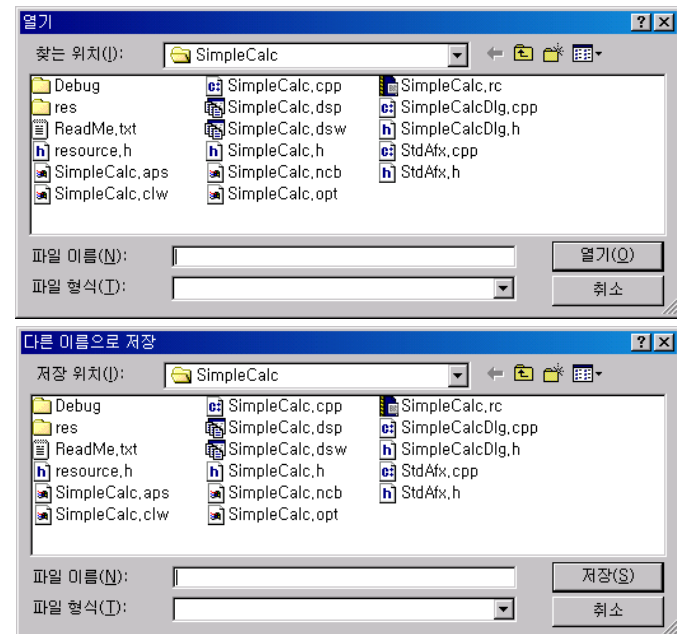
```
CColorDialog dlg(RGB(255, 0, 0), CC_FULLOPEN);  
dlg.DoModal();  
COLORREF color = dlg.GetColor();
```



- CFileDialog

```
CFileDialog dlg(TRUE);  
if(dlg.DoModal() == IDOK)  
    MessageBox(dlg.GetPathName());
```

```
CFileDialog dlg(FALSE);  
if(dlg.DoModal() == IDOK)  
    MessageBox(dlg.GetPathName());
```



Common Dialog Box

- CFontDialog

```
CFontDialog dlg;

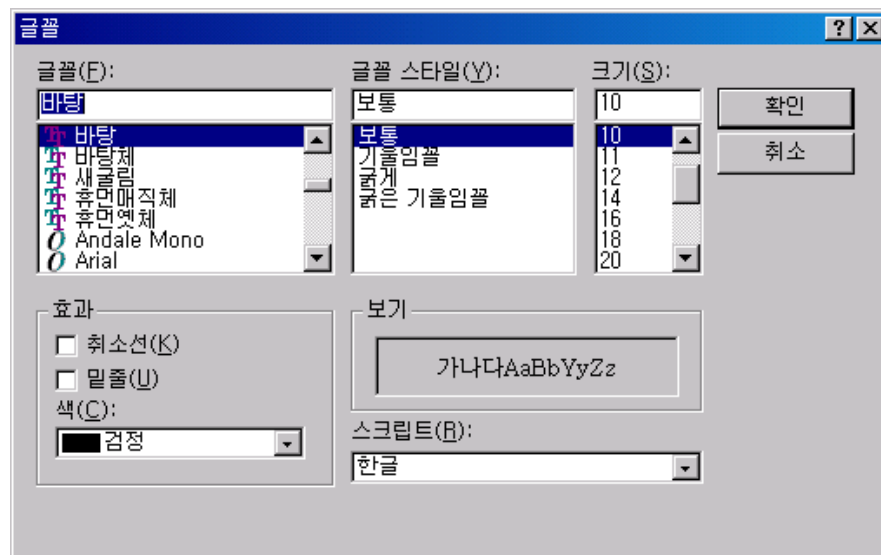
If (dlg.DoModal() == IDOK) {
    CClientDC dc(this);

    // 화면을 지운다.
    CRect rect;
    GetClientRect(&rect);
    dc.SelectStockObject(WHITE_PEN);
    dc.SelectStockObject(WHITE_BRUSH);
    dc.Rectangle(&rect);

    // 선택된 색상을 알아낸다.
    COLORREF color = dlg.GetColor();
    dc.SetTextColor(color);

    // 선택된 폰트를 알아낸다.
    LOGFONT lf;
    dlg.GetCurrentFont(&lf);
    CFont font;
    font.CreateFontIndirect(&lf);
    dc.SelectObject(&font);

    // 텍스트를 출력한다.
    dc.TextOut(10, 10,
        CString("한글 & English"));
}
```

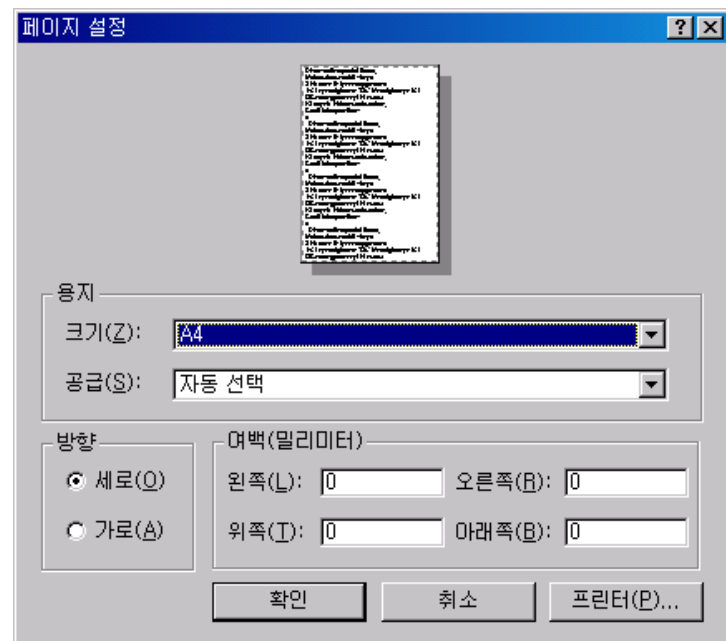
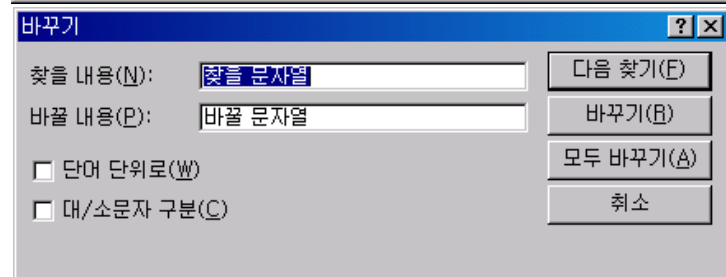
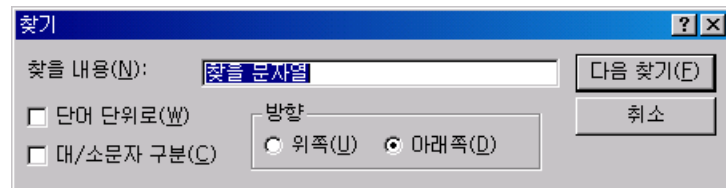


Common Dialog Box

- CFindReplaceDialog
 - 사용 방법은 교재 참조

- CPageSetupDialog

```
CPageSetupDialog dlg;  
dlg.DoModal();
```

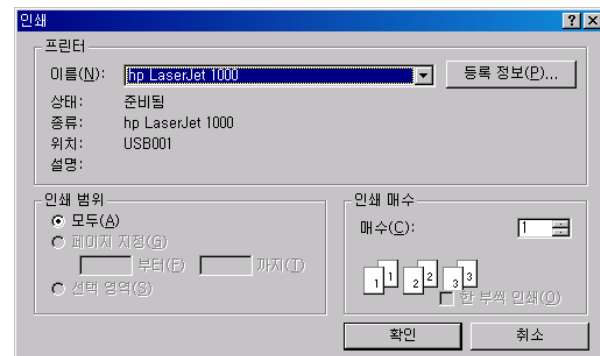
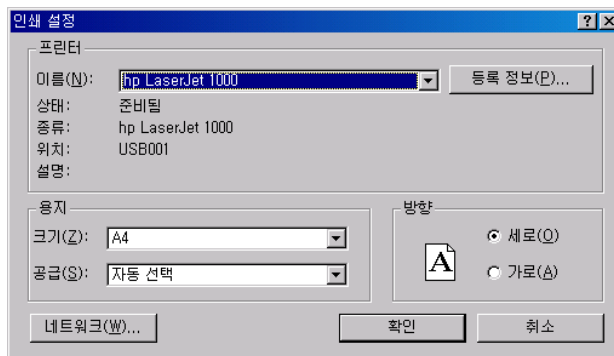


Common Dialog Box

- CPrintDialog

```
CPrintDialog dlg(TRUE);  
dlg.DoModal();
```

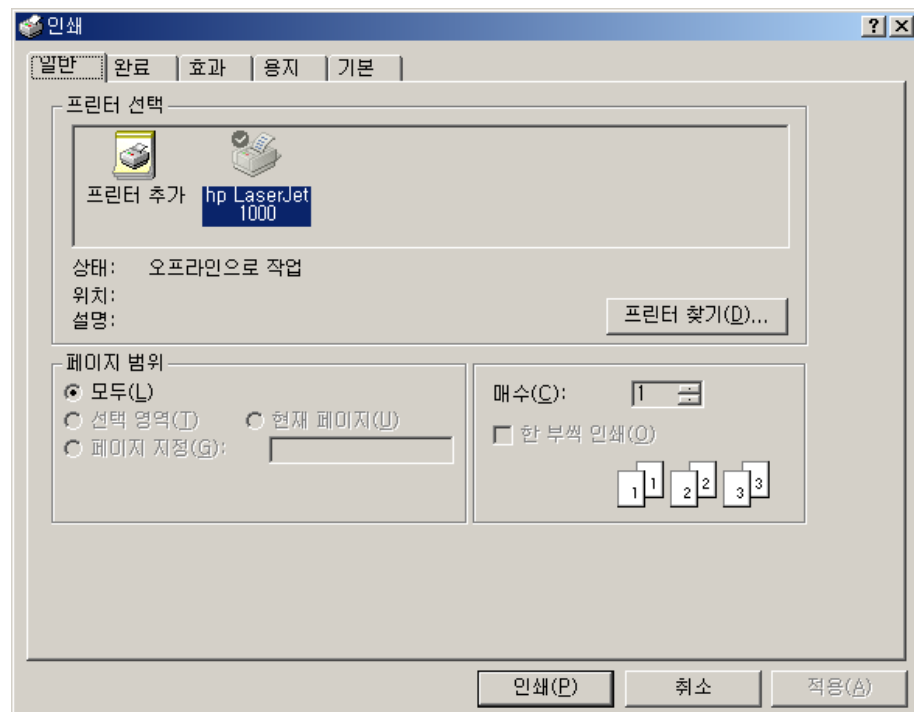
```
CPrintDialog dlg(FALSE);  
dlg.DoModal();
```



- CPrintDialogEx
 - WINVER *0x0500 이상만 사용
 - *Windows 2000

```
// stdafx.h에서 #define WINVER 0x0500 이상 필요
```

```
CPrintDialogEx dlg;  
dlg.DoModal();
```



Q & A